REAPONSE TO CLAIM REJECTIONS

All claims were rejected as obvious, based on Yates and Official Notice.

Applicant points out that the rejections of claims 2 - 13 are defective, because no teaching has been given in those claims for combining Official Notice with Yates.

Also, those rejections are defective under section 103, because the Office Action has not identified the **differences** between those claims and the prior art, as required by Graham v. Deere and MPEP § 706.02(j), which states:.

> Contents of a 35 U.S.C. 103 Rejection
>
> . . . After indicating that the rejection is under 35 U.S.C. 103, the examiner should set forth in the Office action:
>
> . . .
>
> (B) the difference or differences in the claim over the applied reference(s),

Claim 1

Claim 1 recites:

> 1. A method of constructing a plurality of software systems, comprising the following steps:
>
> a) maintaining an inventory of software modules, which includes:
>
> i) a group of type A modules; and
>
> ii) a collection of type B modules;

b) when constructing each software system,

i) including copies of the entire group of type A modules;

ii) including copies of some or all type B modules;

and

iii) generating at least one customized module, which is a copy of neither a type A nor a Type B module.

Claim 1(b)(i) and (ii)

The Office Action relies on

(1) Official Notice

and

(2) Yates, column 18, lines 14 - 26

to show claim 1(b)(i) and (b)(ii).

These claim passages state that

1) **all** of the type A modules are included in the software system being constructed

and

2) **some or all** of the type B modules are included.

The rationale used by the Office Action is that

(1) Yates shows "including copies"

and

(2) Official Notice shows copying both the entire group of A-type modules and B-modules.

However, this rationale utterly fails to comply with section 103, for several reasons.

REASON 1

One reason is that Yates is cited to show "including copies." but the Yates passage cited by the PTO does not show that. The Yates passage refers to a "coordinator 303." The "coordinator" resolves conflicts in Yates system, as when two programs try to gain access to a single piece of data (called an object.) (Column 18, lines 24 - 29.)

The Yates passage has no relevance whatsoever to claim 1. Claim 1 recites a method of constructing a plurality of software systems. The Yates passage has no relevance to that.

REASON 2

A second reason is that merely "including copies" is insufficient, even if Yates showed that, which he does not.

Claim 1(b) states that certain "copies" are included **"when constructing each software system."** That "construction" has not been shown in Yates.

Applicant requests, under 37 CFR §§ 1.104(c)(2) and 35 U.S.C. § 132, that the PTO specifically identify "constructing each software system" in Yates.

THIRD REASON

A third reason is that the Official Notice does not show what claim 1 recites. The Official Notice is that

> Official Notice is taken that copies of the entire group of Type A modules is old and well known in the computer art.

The same type of Official Notice was given with respect to the claimed Type B modules.

But claim 1 does not recite the **mere existence** of "copies of the entire group of Type A modules." Claim 1 recites **including** the entire group of Type A modules "when constructing each software system."

Therefore, the claimed subject matter has not been shown in the prior art, even if Officially Noticed.

As to Official Notice, the undersigned attorney respectfully traverses the Official Notice, and requests a citation of evidence showing the Noticed subject matter. (See MPEP § 2144.03.)

REASON 4

A fourth reason is that the Officially Noticed subject matter does not correspond to the claimed subject matter, for another reason. Under claim 1(a), the "modules" (Type A and B) are "maintained" in "an inventory." Then, in claim 1(b), copies are made "when constructing each software system."

Even if the Official Notice be accepted, that Notice only asserts that "copies of the entire group of type A modules" is well known. But the Notice has not shown that the Noticed copies are "maintained" in the claimed "inventory."

REASON 5

A fifth reason is that the Official Notice contains no informational content. It is a nonsense statement. Thus, no determination can be made as to whether the Officially Noticed subject matter is accurate.

Again, the Notice is this: "copies of the entire group of type A modules" is well known. How would this be verified ?

First, we would find a group of Type A modules. The undersigned attorney has a computer on his desk which contains several software programs. Let us arguendo call them a **group** of Type A modules.

Where are the "copies of the entire group" ? There are none.

Since this attempt at verification failed, and the undersigned attorney sees no other way to verify the Official Notice, it is requested that an explanation be given as to how the truth of the

Official Notice can be verified.

Claim 1(a)

POINT 1

Claim 1(a) recites maintaining type A and type B modules. The Office Action relies on identical passages in Yates to show both of these module types. Those passages are

1) The Abstract

2) Column 2, lines 57 - 65,

3) Column 4, lines 3 - 12,

4) Column 5, lines 40 - 55, and

5) Column 18, lines 1 - 13.

As to item (1), Yates' Abstract merely refers to selecting "reusable software modules." There is no reference to anything analogous to the claimed types A and B.

As to item (2), the Yates passage merely refers to selecting a "set" of software modules.

Item (3) of Yates merely refers to "adding" or "modifying" software modules.

Item (4) of Yates refers to changing a set of software modules which is used at a given time.

Item (5) of Yates states:

> Known constructions, where policies[1] are embedded in the objects, require rewriting of code in the object to change behavior.
>
> External policies allow not only changes in behavior to be achieved more easily but also more freely, and can allow extra behaviors (which are composed from combinations/permutations of a programmed set of operations) to be performed even if these were not originally anticipated.
>
> The concept of policies is such that an object must have access to a "Policy Interpreter."
>
> This can be internal or external to the object.
>
> In order to locate policies, a policy server might be provided, again either internal or external to an object.
>
> (Yates, column 18, lines 1 - 13.)

This passage plainly does not show claim 1(a), which is repeated here:

> a) maintaining an inventory of software modules, which includes:
>
> i) a group of type A modules; and
>
> ii) a collection of type B modules.

POINT 2

The Office Action, page 4, apparently asserts that the

[1] "Policy" is apparently jargon for a section of computer code.

"intelligent agents" in Yates qualify as the Type A and B modules. However, the Office Action has not shown **two different types** of intelligent agents. Thus, claim 1(a) has not been shown in Yates.

Claim 1(b)(iii)

POINT 1

The Office Action attempts to show this claim passage by Official Notice. In response, the undersigned attorney respectfully traverses the Official Notice, and requests a citation of evidence showing the Noticed subject matter. (See MPEP § 2144.03.)

One reason is that the Noticed subject matter contradicts previous Notices. That is, previously it was noticed that

> "copies of the entire group of type A modules"
>
> is well known,
>
> and that
>
> "copies of the entire group of type B modules"
>
> is well known.

Now, it is Noticed that "copies of **neither** a type A nor a type B module" is well known. That is contradictory to the previous two Notices.

POINT 2

The Noticed subject does not correspond to claim 1(b)(iii).

That claim passage refers to an operation occurring "b) when constructing each software system."

That operation has not been shown in Yates.

POINT 3

Claim 1(b)(iii) recites "generating at least one customized module." That has not been shown in the applied art, or even noticed.

At best, the Office Action has shown that type A and type B have not been copied in the prior art.

<u>No Teaching Given</u>

No teaching has been given for combining the Noticed subject matter with Yates. A teaching is required.

POINT 1

The rejection fails to show that the "subject matter **as a whole**" is obvious, as required by section 103.

> -- The rejection asserts that the copying of claim 1(b) is obvious because it provides "backup."
>
> -- The rejection asserts that the **non-copying** of claim 1(b)(iii) is obvious because it provides "flexibility."

That, even if valid reasoning, is piecemeal combination, and is not allowed.

From another point of view, at best, that reasoning shows that

-- claim 1(b)(i) and (ii) are obvious, by themselves, and

-- claim 1(b)(iii) is obvious, by itself.

But it fails to show that all of claim 1(b) is obvious, **as a whole**.

It fails to show that the combination of 1(b)(i), 1(b)(ii), and 1(b)(iii) is obvious. It only shows, if valid, that sub-combinations of these three elements are obvious.

POINT 2

The rationales used do not actually lead to the claimed invention.

-- If "backup" is desired, then one does not "construct" a "software system," as in claim 1(b). One merely copies what one wishes to back up.

-- Claim 1(a) recites maintaining an inventory of software modules. Everybody knows that some type of back-up is used when storing software, unless the storage media being used are clearly indestructible. Thus, if back-up is desired, it occurs in claim

> 1(a), and there is no reason to assert that claim 1(b) is done for back-up purposes.

POINT 3

The invocation of "flexibility" is insufficient as a rationale under section 103, for several reasons.

One reason is that no definition of flexibility has been given.

A second is that no facts have been given which prove that non-copying of A and B provide flexibility.

A third reason is that the rationale is a logical impossibility. The rationale states that **doing nothing** increases flexibility. That makes no sense.

Claim 2

Claim 2 recites:

> 2. Method according to claim 1, wherein each system constructed performs the following functions:
>
> 1) processing of the content of messages;
>
> 2) packaging of messages into packets for transport out of the system;
>
> 3) transfer of messages into, and out of, the system; and
>
> 4) coordination of functions (1), (2),

and (3).

CLAIM 2(1)

The Office Action relies on five passages in Yates to show claim 2(1), namely:

column 1, lines 1 - 5;

column 2, lines 57 - 65;

column 4, lines 3 - 12;

column 16, lines 11 - 23; and

column 5, lines 33 - 35.

However, claim 2(1), together with its context, states that

> . . . each system constructed performs the following functions:
>
> 1) processing of the content of messages.

The first passage cited in Yates says nothing about whether "each system constructed" performs a specific function, let alone the claimed function of processing messages.

The second passage of Yates states that "A reconfigurable software agent **MAY** comprise . . ." That is inconsistent with claim 2(1). If claim 1(1) were shown in that passage, then the passage would state that the reconfigurable software agent "**MUST** comprise," or equivalent.

Restated, claim 2(1) is **mandatory**. Each system constructed

must perform what claim 2(1) states.

The third passage of Yates merely lists software modules which **may** be available. Again, that does not show claim 2(1).

The fourth passage of Yates merely lists some functions which a "service retailer domain 103" performs. That does not show claim 2(1).

The fifth passage of Yates merely discusses reconfiguration of his software modules. That does not show claim 2(1).

Therefore, the mandatory language of claim 2 ("**each** system constructed performs the following functions . . .") has not been shown in Yates.

CLAIM 2(2)

The Office Action relies on two passages in Yates to show claim 2(2), namely:

column 14, lines 37 - 51; and

column 18, lines 9 - 13.

Claim 2(2) recites:

> . . . wherein **each system constructed** performs the following functions:
>
> . . .
>
> 2) packaging of messages into packets for transport out of the system;

The first cited passage in Yates merely refers to a type of communication. But it does not state that "each system constructed . . ." possesses that communication. Thus, even if the type of communication shown in Yates corresponds to claim 2(2), claim 2(2) cannot be read out of context.

The second passage of Yates states:

> The concept of policies is such that an object must have access to a "Policy Interpreter." This can be internal or external to the object. In order to locate policies, a policy server might be provided, again either internal or external to an object.
>
> (Column 18, lines 9 - 13.)

The second passage clearly does not relate to claim 2(2).

Therefore, the mandatory language of claim 2 ("**each** system constructed performs the following functions . . .") has not been shown in Yates.

CLAIM 2(3)

To show claim 2(3), the Office Action cites Yates, column 5, lines 40 - 55. Claim 2(3) recites:

> . . . wherein each system constructed performs the following functions:
>
> . . .
>
> 3) transfer of messages into, and out of, the system.

The cited passage in Yates says absolutely nothing about transferring messages into and out of the system, together with requiring that "each system constructed" does that transfer.

The Office Action also cites the Abstract of Yates. The Abstract simply does not show claim 2(3).

The Office Action also cites Yates, column 17, lines 27 - 37. That passage has no relevance whatever to the claim passage in question. That passage discusses how Yates responds to a type of incoming data.

Therefore, the mandatory language of claim 2 ("each system constructed performs the following functions . . .") has not been shown in Yates.

CLAIM 2(4)

Claim 2(4) recites:

> . . . wherein each system constructed performs the following functions:
>
> . . .
>
> 4) coordination of functions (1), (2), and (3).

As explained above, claim 2(1), (2), and (3) are not found in Yates. Thus, the passage in question, namely, column 4, lines 3 - 12, cannot refer to "coordination" of the functions of those parts of the claim.

In addition, the passage in question (column 4, lines 3 - 12) does not refer to the other passages cited in Yates to show the other parts of claim 2. Thus, it would seem conclusive that the passage in question (column 4, lines 3 - 12) does not show the claimed coordination, since it does not refer to coordinating elements in the other passages cited.

Therefore, Applicant submits that claim 2 is not shown in Yates.

ADDITIONAL POINT

The rejection was stated to be obviousness-type. (Office Action, page 3.) But only a single reference was used, and is alleged to show all claim elements.

The rejection is invalid.

Claim 3

Claim 3 recites:

> 3. Method according to claim 2, wherein functions (3) and (4) are performed using **type A modules** exclusively.

"Type A modules" are those recited in claim 1. Claim 1 states that **all** type A modules in the group are included in every system constructed.

Claim 3 depends from claim 2, which defines "functions (2) and

(3)." The passages cited in Yates to show those functions (2) and (3) of claim 2 do not refer to "type A modules."

In addition, one passage used to reject claim 3 is column 18, lines 1 - 13. That passage of Yates relied on by the PTO is here set forth:

> Known constructions, where policies[2] are embedded in the objects, require rewriting of code in the object to change behavior.
>
> External policies allow not only changes in behavior to be achieved more easily but also more freely, and can allow extra behaviors (which are composed from combinations/permutations of a programmed set of operations) to be performed even if these were not originally anticipated.
>
> The concept of policies is such that an object must have access to a "Policy Interpreter."
>
> This can be internal or external to the object.
>
> In order to locate policies, a policy server might be provided, again either internal or external to an object.
>
> (Yates, column 18, lines 1 - 13.)

Plainly, this passage does not show claim 3.

The Office Action also relies on Yates, column 17, lines 16 - 21. That passage was also used to reject claim 2(4), which is different from claim 3.

[2] "Policy" is apparently jargon for a section of computer code.

That passage merely states that an "agent" is constructed of three "object types." That does not show claim 3.

Claim 4

Claim 4 recites:

> 4. Method according to claim 3, wherein function (1) is performed using a combination of type A, type B, and customized modules.

The Office Action relies on two passages in Yates to show claim 4, namely,

column 4, lines 26 - 35 and

column 18, lines 1 - 13.

The latter passage was cited in full in the previous section, concerning claim 3, and plainly does not show claim 4, as a side-by-side comparison indicates.

The former passage states that "at least one of the software agents" is provided with certain functionality. If the "software agent" is considered to correspond to the "system" of the claims, and the undersigned attorney sees no other element in the passage which can correspond to the "system," then claim 4 is clearly not shown. Claim 4, through its parent claim(s), states that "each system constructed" is equipped with the enumerated functions. The passage in question merely states that "at least one software agent" is equipped with certain functions.

That does not show claim 4.

Claim 5

Claim 5 recites:

> 5. Method according to claim 4, wherein function (2) is performed using a combination of type A, type B, and customized modules.

The identical passages in Yates cited to show claim 4 were cited to show claim 5.

One passage is cited in full, in the section above regarding claim 3. A side-by-side comparison clearly indicates that the passage does not show claim 5.

The other passage (column 4, lines 26 - 35) states that "at least one of the software agents" is provided with certain functionality. If the "software agent" is considered to correspond to the "system" of the claims, and the undersigned attorney sees no other element in the passage which can correspond to the "system," then claim 5 is clearly not shown.

Claim 5, through its parent claim(s), states that "each system constructed" is equipped with the enumerated functions. The passage in question merely states that "at least one software agent" is equipped with certain functions.

That does not show claim 5.

Claim 6

Claim 6(a)

Applicant points out that claim 6(a) recites:

> a) fabricating a collection of software systems, **each of which contains**

and then lists four types of module (which are "contained").

Restated, claim 6(a) states that every "software system" in the "collection" contains (at least) the four modules listed in 6(a)(i) through (a)(iv).

The Office Action cites two passages in Yates to show this, namely,

1) column 2, lines 38 - 65

and

2) column 4, lines 13 - 65.

However, those passages contain nothing more than generalized statements indicating that, in different situations, systems may be designed which are different.

That is **directly contrary** to the claim recitations in question. One reason is that claim 6(a) states that every software system contains four specific modules. Thus, in that respect, every software system is **identical**.

That is contrary to the cited passages in Yates.

Claim 6(b)

Claim 6(b) states that all of the "software systems" of claim 6(a) will contain two elements, namely (1) identical CONTROL modules and (2) identical COM_MOD modules. The Office Action relies on two passages in Yates to show this.

One passage is column 4, lines 26 -35. However, that passage merely states that "at least one . . . software agent" is equipped with certain functionality. That does not show the claim recitations in question, and is actually inconsistent with it.

It is inconsistent because the Yates-passage only focuses on **ONE** software agent, and lists some properties of that agent. It fails to identify a **GROUP** of agents. Claim 6 refers to properties of a "collection of software systems." Yates' discussion of a **SINGLE** software agent does not show the properties of a **GROUP** as in claim 6.

The other passage relied on by the PTO is Yates column 18, lines 1 - 13. That passage is set out verbatim above, and clearly does not show claim 6(b)(i) and (b)(ii).

Claim 6(b)(iii)

Applicant points to claim 6(b)(iii), which is repeated here:

> iii) fabricating PAK_MOD modules in all systems, such that:

A) copies of a software unit A is contained in every PAK_MOD module;

B) some PAK_MOD modules contain a software unit B with no unit C; and

C) some PAK_MOD modules contain a software unit C with no unit B.

The undersigned attorney has examined the passages in Yates which are cited to show these recitations, and cannot locate the recitations in those passages. [Actually, the passages used by the PTO are the same as used for claim 6(b)(i) and (b)(ii).]

Further Consideration of Claim 6

Claim 6(a)(iii) refers to a "communications module (COM_MOD) which accepts and delivers message packets." Yates, column 14, line 49 et seq., refers to a Manager which uses data packets for file transfer. It is Assumed arquendo that Yates' Manager shows the recited COM_MOD.

Claim 6(b)(ii) states that, during fabrication of software systems, "fabricating identical COM_MOD modules in all systems." The Office Action relies on two passages in Yates to show this.

One passage is column 4, lines 26 - 35. However, that passage, in essence, states that each "software agent" is "customized" for a specific purpose. Thus, the agents will be different. That does not state, or even imply, "identical COM_MOD modules in all" agents.

The second passage is column 18, lines 1 - 13. That passage was set forth verbatim above. That passage merely refers to a process of modifying software. That does not state, or even imply, "identical COM_MOD modules in all" agents.

In fact, it would tend to show the opposite. If software (ie, the COM_MOD modules) is modified, then for "identical COM_MOD modules" to exist in all agents, all those modules must be modified in **the same manner**. That has not been shown in Yates.

Claim 7

Claim 7 recites:

> 7. Method according to claim 6, and further comprising the following step:
>
> iv) fabricating PROC_MOD modules in all systems, such that:
>
> A) copies of a software unit D is contained in every PROC_MOD module;
>
> B) some PROC_MOD modules contain a software unit E with no unit F; and
>
> C) some PROC_MOD modules contain a software unit F with no unit E.

The same two passages in Yates, cited to show claims 4 and 5 were cited to show (A), (B), and (C) of claim 7(iv). Those passages are

column 4, lines 26 - 35 and

column 18, lines 1 - 13.

The latter passage is here repeated:

> Known constructions, where policies[3] are embedded in the objects, require rewriting of code in the object to change behavior.
>
> External policies allow not only changes in behavior to be achieved more easily but also more freely, and can allow extra behaviors (which are composed from combinations/permutations of a programmed set of operations) to be performed even if these were not originally anticipated.
>
> The concept of policies is such that an object must have access to a "Policy Interpreter."
>
> This can be internal or external to the object.
>
> In order to locate policies, a policy server might be provided, again either internal or external to an object.
>
> (Yates, column 18, lines 1 - 13.)

A side-by-side comparison between each recitation in claim 7 and the cited passage clearly shows that the cited passage does not show the claim elements. For example, claim 7(iv)(C) states:

> C) some PROC_MOD modules contain a software unit F with no unit E.

The absence of the unit E is not discussed in this passage. That,

[3] "Policy" is apparently jargon for a section of computer code.

by itself, is sufficient to preclude the rejection.

The other passage (column 4, lines 26 - 35) states that "at least one of the software agents" is provided with certain functionality. If the "software agent" is considered to correspond to the "system" of the claims, and the undersigned attorney sees no other element in the passage which can correspond to the "system," then claim 7 is clearly not shown.

Claim 7, through its parent claim(s), states that "each system constructed" is equipped with the enumerated functions. The passage in question merely states that "at least one software agent" is equipped with certain functions.

That does not show claim 7.

General Observations on Yates

If an attempt is made to apply Yates to claim 1, the undersigned attorney believes that the only possible elements in Yates which apply to (1) the "software system," (2) the type A modules, and (3) the type B modules of claim 1 are the following, respectively:

-- Yates' "agents" (corresponding to the "software system"),

-- Yates' SIBBs, Service Independent Building Blocks (corresponding to the type A modules), and

-- Yates' "adaptors" (corresponding to the type B modules.)

(See column 17, lines 12 - 22.)

However, several problems arise. Claim 1(b)(i) states that "the entire group" of the type A modules is included in the "software system." Yates expressly states that is not so. He states that the SIBBs in the "agents" change over time. (Column 18, lines 42 - 48.) Thus, as a minimum, the "entire group" of the SIBBs does not remain constant.

And the undersigned attorney can find no statement that "the entire group" of the SIBBs is given to each "agent" in the first place. Again, claim 1 is not present in Yates.

Another problem arises in connection with claim 3, which states:

> 3. Method according to claim 2, wherein functions (3) and (4) are performed using type A modules exclusively.

"Functions (3) and (4)" of claim 2 refer to transfer of messages into, and out of, the system. Yates' SIBBs do not do that. His Communications Session Manager does that. (Column 14, line 49 et seq.) That Manager does not appear to be a SIBB.

Therefore, Applicant requests that the "group of type A modules" be identified in Yates.

Claims 8 - 11

All these claims recite installing "the software system," which was "constructed," into an "electronic payment switch." The term "electronic payment switch" does not appear in Yates.

Thus, the PTO must provide an explanation as to how the elements relied on in Yates correspond to the claimed "electronic payment switches."

Claims 12 and 13

Claim 12 recites:

> 12. Method according to claim 1, and further comprising:
>
> c) repeating steps of paragraph (b) to thereby modify a software system previously constructed; and
>
> d) installing the modified software system into an electronic payment switch.

The Office Action asserts that the fact that something in Yates is "reusable" and has "reconfigurability" shows claim 12(c). (Office Action, page 9.) However, that is insufficient to show claim 12(a). One reason is that the entity in question in Yates has not been identified.

Another reason is that even if the entity in Yates corresponds to the claimed "software system," the fact that this system in Yates is "reusable" and has "reconfigurability" does not show claim

12(a).

-- Claim 12(a) states that specific steps are **repeated**. It is not necessary to repeat those steps to attain something which is "reusable" and has "reconfigurability."

-- Restated, you can attain "reusability" and "reconfigurability," in other ways. For example, you can increase horsepower in a car by boring out the cylinders. You can repeat that boring, to further increase horsepower.

On the other hand, you can increase horsepower in a completely different way, as by adding a turbocharger.

Therefore, in general, you can "reconfigure" a device in different ways. The fact that a reference discusses reconfiguration does not imply a specific way of reconfiguring.

This applies to claim 13.